3. Issues for the Selection and Development of Models and Computer Codes

3.1 INTRODUCTION

In assessing the containment characteristics of the WIPP disposal system, DOE will need to consider many complex processes upon which their decisions regarding performance assessment will be based. Although some decisions can be made using engineering judgment, many analyses must be performed where human reasoning alone is inadequate to synthesize the many factors involved in complex problems. The best tool available to help scientists meet the challenge of such analyses and predictions is a model.

A model is a system designed to represent a simplified version of a real system. Models, can be valuable predictive tools for performance assessment if properly constructed. The validity of the predictions will depend on how well or conservatively the model approximates the physical system being modeled.

DOE is currently developing models and computer codes to meet performance assessment objectives for the WIPP. EPA will ultimately accept or reject DOE's selection and application of models and computer codes. This section identifies the issues considered by EPA in the development of criteria for WIPP model and code selection, description, implementation, and justification (§194.23). These criteria are based on full implementation of quality assurance procedures and the complete documentation of the procedures.

DOE has already selected a number of computer codes at the WIPP to gain an insight into the kinds of problems that may be encountered in the modeling analyses that will be conducted for the performance assessment. In the process of DOE's continued formulation and testing of the various components of the disposal system conceptual models, attention is given to many aspects of the system and possible avenues of analysis. This includes developing a conceptual model of the site that defines the physical framework, relevant processes, boundary conditions, and what approaches are justifiable and relevant to meeting performance objectives. After the conceptual model has been formulated, appropriate codes are selected by matching a detailed description of the modeling needs with well-defined, quality-assured characteristics of existing codes, while taking into account the compliance assessment objectives of the study. If a good match between model requirements and code characteristics cannot be found, modification of an existing code or the development of a new code may be considered.

In the 1992 PA, DOE presented components of its site conceptual model. These components focused on the Culebra Dolomite Member of the Rustler Formation overlying the Salado Formation. The Culebra Dolomite is thought to present the most likely avenue for radioactive waste to reach the accessible environment in addition to direct releases to the accessible environment. However, there is considerable uncertainty regarding the physical system boundary conditions and flow and transport mechanisms.

Another consideration is that computer codes are generally not designed to be universally applicable. Code development is normally aimed at solving a specific environmental problem or range of problems. Therefore, a single code will not simulate all of the components of the conceptual model. For example, most WIPP performance assessment scenarios consider three time-dependent gas-generation processes which are expected to be involved in the degradation of transuranic (TRU) waste in the disposal system: (1) oxic and anoxic corrosion of metals, (2) aerobic and anaerobic microbial degradation of cellulosic materials, and (3) radiolysis. The potential for large quantities of gas has strong links to other processes associated with closure of the disposal rooms and panels. After the repository is closed, the surrounding halite will close (creep) inward upon and compact the waste and backfill. Gas generating processes in the waste and the impact of creep closure will potentially increase pressure in the room. The pressure within these materials may force the brine and compressed gas through natural and induced fractures toward the regulatory boundaries.

Section 3.2 of this chapter discusses how a code review would first determine if the selected code was compatible with the modeling objectives set forth in the performance assessment. Section 3.3 focuses on the code development process, code capabilities, and the quality of the accompanying documentation. Issues related to the application of a computer code are considerably different than those associated with code development and selection (i.e., a properly selected code can be improperly applied). Although the primary objective of this chapter is to present issues related to model and code selection rather than code application, Section 3.4 discusses code application criteria in a global sense.

3.2 REVIEW OF COMPUTER CODE BY EVOLUTION EVALUATION

Computer code selection is the process of choosing the appropriate software capable of simulating the characteristics of the physical system to be modeled. The evolution of the computer code can be traced from the inception of the conceptual model to the formulation of the mathematical model, and finally to the development of the computer code where computer instructions for performing the operations specified in the mathematical model are programmed. The formulation of a conceptual model is an integral component of the modeling process. Components of the conceptual model may be simplified to meet either limited objectives or limitations in the data. It is often useful to simulate only certain components of the conceptual model. For instance, even if there are data that indicate different geologic zones in the hydrogeologic unit, it is common practice to evaluate the system as a function of average values. While different aspects of the conceptual model may be simulated in a variety of ways, the selected approach must remain consistent with the objectives. The review and acceptance of a model is an evolutionary process that depends upon the modeling goals and availability of data. The following illustration is focused on hydrogeologic models although the process would be nearly identical for other models.

3.2.1 Conceptual Models

The performance assessment will use numerous conceptual models to describe the physical, biological and chemical processes expected to occur at WIPP. At the most basic level, conceptual models describe very fundamental processes, for example, the type of microorganisms present in the repository, their population size and metabolic rates. These basic conceptual models are integrated further to form the basis for more complete conceptual models that predict processes such as gas generation, room closure rates, and contaminant transport mechanisms.

The conceptual model of a groundwater system is an interpretation of the characteristics and dynamics of the physical hydrogeologic system. The purpose of the conceptual model is to consolidate site and regional hydrogeologic and hydrologic data into a set of assumptions and concepts that can be evaluated quantitatively. The system conceptualization should include: the geologic and hydrologic framework, characteristics of geologic formations (e.g., fractured or porous), the nature of relevant physical and chemical processes, time dependent processes, geometry of the system, initial and boundary conditions, hydraulic properties; and sources and sinks (water budget). A brief discussion of each of the typical components of the hydrogeological conceptual model that should be considered and reviewed is presented below.

Geologic framework. The geologic framework is the distribution and configuration of various rock units (e.g., fractured dolomite or intact halite). Of primary interest are the thickness, continuity, lithology, and geologic structure of those units that are relevant to the purpose of the study.

Hydrologic framework. The hydrologic framework in the conceptual model includes the physical extent of the flow system, hydrologic features that impact or control the groundwater flow system, analysis of groundwater flow directions, and media type. The conceptual model must address the degree to which the system behaves as a porous media. If the system is significantly fractured or solution channeled, the conceptual model must address these issues.

Hydraulic properties. The hydraulic properties include the transmissive and storage characteristics of the rocks and properties of the fluids. Specific examples of rock and fluid properties include transmissivity, hydraulic conductivity, storativity, fluid viscosity and densities. Hydraulic conductivity may also have directionality (anisotropy).

Sources and sinks. Sources or sinks of water or gas impact the pattern and rate of flow and may affect the transport of radionuclides from the repository. The most common examples of sources and sinks include pumping or injection wells, infiltration, evapotransporation, drains and flow from surface water bodies. At the WIPP reactions between brine and waste in the repository may provide a source of gas.

Boundary and initial conditions. Boundary conditions are the conditions the modeler specifies, typically on the perimeter of the model domain, in order to solve for the unknowns in the problem domain. These values may be associated with either the groundwater flow or the contaminant transport aspects of the problem. Groundwater boundaries may be described in terms of where water and/or gas are flowing into the groundwater system and where water and/or gas are flowing out. Many different types of boundaries exist, including: surface water bodies, groundwater divides, rainfall, wells, and geologic features such as faults and sharp contrasts in lithology. For example, at the WIPP, pressure boundary conditions for the Salado have been set to far field fluid pressures.

The most common contaminant-source boundaries specify the source concentration or prescribe the mass flux of contamination entering the system. Both type of source boundaries are currently used in the modeling at the WIPP.

Initial conditions are defined as values of groundwater elevation, pressure, flow volumes, or contaminant concentrations which are initially assigned to interior areas of the modeled regions. At the WIPP, pressures in the repository are initially set to atmospheric conditions.

Transport processes. The transport of radionuclides by flow through either a porous matrix or a fracture system in a porous matrix will be affected by various mechanical and geochemical processes. The dominant mechanical processes are advection, dispersive effects (hydrodynamic dispersion, channeling) and diffusion. The chemical processes potentially affecting radionuclide transport include: adsorption on mineral surfaces (both internal and external to the crystal structure), speciation, precipitation, colloidal transport, radiolysis, biofixation, natural organic matter interactions, anion exclusion, and complexation.

Spatial dimensionality. Groundwater flow and contaminant transport are seldom constrained to one or two dimensions. However, in some instances, it may be appropriate to restrict the analysis to one or two dimensions. The decision to model a site in a particular number of dimensions should be based upon the modeling objectives and the availability of field and/or laboratory data.

Temporal dimensionality. Either steady-state or transient flow simulations can be performed. At steady-state, it is assumed that the flow field remains constant with time, whereas a transient system simply means one that changes with time. Steady-state simulations produce average or long-term results and require that a true equilibrium case is physically possible. Transient analyses are typically performed when boundary conditions are varied through time or when study objectives require answers at more than one time.

The conceptual model is based on the modeler's experience, technical judgment and represents the modeler's understanding of the physical system being modeled. The conceptual model will become more complex as more processes are identified and interrelationships of important components within the systems are considered. The transformation of the conceptual model into a mathematical model is only an extrapolation of a basic understanding of the system will result in simplifications of the system. For example, the mathematical models assume that there is a direct scaling between the model simulations and the scale at which the data are collected. The lack of knowledge about the system resulting from limited information also contributes to simplifications of the mathematical models.

In addition to the unavoidable simplifications of the conceptual model, there are simplifications in which the modeler decides what physical characteristics and processes are important to the model application. Examples of these kind of simplifying assumptions include:

- Flow through the unsaturated zone is vertical and in one dimension.
- Chemical reactions are reversible and instantaneous.
- Soil or rock medium is isotropic and/or homogeneous.
- Flow field is uniform and under steady-state conditions.

As more data become available, simplifying assumptions are removed and the conceptual model complexity increases. This process creates mathematical model development which allows for the systematic integration of previously neglected conceptual model components.

3.2.2 Mathematical Models

A conceptual model describes the present condition of the system. To make predictions of future behavior it is necessary to develop mathematical models. Laboratory sand tanks are physical scale models that simulate groundwater flow directly. The flow of groundwater can also be implied using electrical analog models. Mathematical models, including analytical and numerical methods, discussed below, are more widely used because they are easier to develop and manipulate.

A mathematical model is essentially a mathematical representation of a process or system conceptual model. For example, the mathematical model for groundwater flow is derived by applying principles of mass conservation and conservation of momentum. The generally applicable equation of motion in groundwater flow is Darcy's law for laminar flow, which originated in the mid-nineteenth century as an empirical relationship. Later, a mechanistic approach related this equation to the basic laws of fluid dynamics. In order to solve the flow equation, both initial and boundary conditions are necessary.

Solution Methodology

Every groundwater model is based upon a set of mathematical equations. Solution methodology refers to the strategy and techniques used to solve these equations. The equations are normally solved for water elevations in the subsurface (head) and/or contaminant concentrations.

Mathematical methods developed to solve groundwater flow and transport equations can be broadly classified as either deterministic or stochastic. Deterministic methods assume that a system or process operates such that the occurrence of a given set of events leads to a uniquely definable outcome. Stochastic methods presuppose the outcome to be uncertain and are structured to account for this uncertainty.

Most of the stochastic methods are not purely stochastic because they often utilize a deterministic representation of soil processes and derive their stochastic nature from their representation of inputs and/or spatial variation of soil characteristics and resulting chemical movement. While the deterministic approach results in a specific value of a soil variable (e.g., solute concentration) the stochastic approach provides the probability of a specific value occurring at any point.

The development of stochastic methods for solving groundwater flow is a relatively recent endeavor that has occurred as a result of the growing awareness of the importance of intrinsic variability of the hydrogeologic environment. Stochastic methods are still primarily research tools; however, as computer speeds continue to increase, stochastic methods will be able to move further away from the research-oriented community and more into mainstream management applications. This discussion focuses primarily on deterministic methods.

Deterministic methods may be classified as either analytical or numerical. Analytical methods usually involve approximate or exact solutions to simplified forms of the differential equations for water movement and solute transport. Simple analytical methods are based on the solution of differential equations which give qualitative estimates of the extent of contaminant transport. Such methods are simpler to use than numerical methods and can generally be solved with the aid of a calculator or computers. Analytical methods are restricted to simplified representations of the physical situations and generally require only limited site-specific input data. They are useful for screening sites and scoping the problem to determine data needs or the applicability of more detailed numerical methods.

Analytical methods are used in groundwater investigations to solve many different kinds of problems. For example, aquifer parameters (e.g., transmissivity, storativity) are obtained from aquifer tests through the use of analytical methods. To avoid confusion, only analytical methods designed to estimate groundwater flow and radionuclide transport rates are discussed in this section.

Analytical methods that solve groundwater flow and contaminant transport in porous media are comparatively easy to use. Analytical solutions are generally restricted to either radial flow problems or to cases where velocity is uniform over the area of interest. Except for some radial flow problems, almost all available analytical solutions are developed for systems having a uniform and steady flow. This means that the magnitude and direction of the velocity throughout the system are invariable with respect to time and space.

Equations of flow and continuity in the form of partial differential equations do not lend themselves easily to rigorous analytical solutions when boundaries are complex. Generally, a realistic analytical expression for hydraulic head or concentration as a function of space cannot be written from the governing equations, boundary and initial conditions, and therefore analytical methods are generally abandoned and replaced by more sophisticated numerical methods.

Numerical methods provide solutions to the differential equations describing water movement and solute transport using methods such as finite differences and finite elements. Numerical methods can account for complex geometry and heterogenous media, as well as for dispersion, diffusion, and chemical retardation processes (e.g., sorption, precipitation, radioactive decay, ion exchange, degradation). These methods always require a digital computer, greater quantities of data than analytical modeling, and an experienced modeler-hydrogeologist.

3.2.3 Numerical Models

A numerical model for groundwater flow consists of the mathematical framework for the solution of the material balance equations that govern laminar flow through porous or fractured media. These mass balance equations are dependent upon physical constraints and constitutive relationships. The constraints simply state conditions that components of the mass balance equations must satisfy, whereas the constitutive relationships describe the dependence of parameters, in the mass balance equations, on other physical processes. Furthermore, the mass balance equations are composed of both spatial and temporal terms both of which require discretization within the model domain. These terms describe the head or pressure in space and time. Either finite element or finite difference methods can be used to discretize the spatial term in the mass balance equations, whereas finite differences are almost always used to discretize the temporal term.

The mass balance equations, physical constraints, and constitutive relationships lead to a series of equations that must be solved in space and time. The means by which the equations are discretized, linearized (e.g. Newton-Raphson), organized (i.e, matrix construction) and solved with either direct or iterative methods is all part of the numerical model.

3.2.4 Computer Code

It is important that the progression from the conceptual model to the computer model is documented in detail. The discussion for each component of the conceptual model should begin with the laboratory or field studies that provide the fundamental characterization data. Next, the data analysis results that support a particular conceptualization should be presented. As part of the data analysis discussion the basis for screening out reasonable alternative conceptual models should be provided. This type of discussion should be presented not only for major components of the conceptual model (e.g., Darcian versus non-Darcian flow) but also for more obscure assumptions (e.g., sorption may be described by a linear isotherm). Following the development of each major and minor component of the conceptual model, the formulation of the mathematical models and numerical models should be presented.

The linkage between the conceptual model(s), mathematical model(s) and numerical model(s) should be clearly described. For example, conceptual models generally assume a continuum in space and time, whereas mathematical models frequently divide space and time into user specified segments. Furthermore, the implications that this type of simplifying assumption may have on the modeling predictions should be presented.

Following the formulation of the numerical model, the computer program is developed. The computer program consists of the assembly of numerical techniques, bookkeeping, and control language that represents the model from acceptance of input data and instructions to delivery of output.

In summary, the conceptual model is a working description of the characteristics and dynamics of a physical system. Model construction is the process of transforming the conceptual model into a non-unique, simplified, mathematical description of the physical system, coded in computer programming language together with a quantification of the simulated system. An intermediate step in the model transformation process is the mathematical model which consists of two aspects: a process equation and a solution technique to solve the process equation. An analytical solution solves a very simple process equation analytically by hand calculations. An analytical model solves a more complex, but still relatively simple, process equation analytically with a computer program. A numerical model solves a simple or complex process equation numerically with a computer program. In the context of this document, mathematical model refers to all three solution techniques of a process equation. The complexity of the process equation dictates the solution technique required. The model formulation process concludes with the coding of the mathematical model into computer programming language for performing a specified set of operations.

3.3 CODE-RELATED ISSUES

The determination of a computer code's acceptability for a particular application at the WIPP depends on whether the code can meet the modeling objectives. The code evaluation process must also consider attributes that are integral components of the computer code(s) including:

- Source Code Availability
- History of Use
- Quality Assurance
- Code Documentation
- Code Testing
- Hardware Requirements
- Solution Methodology

- Code Dimensionality

3.3.1 Source Code Availability

Detailed documentation of the software, source code and developmental history is required to facilitate independent review. Independent evaluation of the reproducibility of the verification and validation results require compiled version of the code (i.e., computer code in machine language) should be available to the reviewer, together with files containing the original test data used in the code's verification and validation.

3.3.2 History of Use

The evaluation process should rely on documented user experience, in addition to hands-on experience and testing. User experience is especially valuable in determining whether the code functions as documented or has significant errors and shortcomings. In some instances users independent of the developer should perform extensive testing and bench-marking.

3.3.3 Quality Assurance

Code acceptance issues are closely tied to the quality assurance procedures followed during the developmental process of the computer code. These criteria are associated with the adequacy of the code testing and documentation.

Quality assurance in modeling is the procedural and operational framework put in place by the organization managing the modeling program, to assure technically and scientifically adequate execution of all project tasks included in the program, and to assure that all modeling-based analysis is verifiable and defensible (TAY85).

The American Society of Mechanical Engineers (ASME) Committee on Nuclear Quality Assurance has developed standards for the development and use of computer software used in the design and operation of nuclear facilities (ASM90). The standard in NQA-2a-1990 Addenda (Part 2.7) was developed under procedures accredited as meeting the criteria for American National Standards. It addresses the following:

- general requirements
- software life cycle

- software verification and validation
- software configuration control
- documentation
- verification reviews
- problem reporting and corrective action
- access control
- software procurement
- records

Quality assurance requirements in 40 CFR part 194 (§194.23(b)) mandate the use of this standard as the basis for documenting any computer codes used to support a compliance application.

The two major elements of quality assurance are quality control and quality assessment. Quality control refers to the procedures that ensure the quality of the final product. These procedures include the use of appropriate methodology in developing and applying computer simulation codes, adequate verification and validation procedures, and proper usage of the selected methods and codes (HEI92). To monitor the quality control procedures and to evaluate the quality of the studies, quality assessment is applied (HEI89).

Software quality assurance (SQA) consists of the application of procedures, techniques, and tools through the software life cycle, to ensure that the products conform to pre-specified requirements (BRY87). This requires that in the initial stage of the software development project, appropriate SQA procedures (e.g., auditing, design inspection, code inspection, error-prone analysis, functional testing, logical testing, path testing, reviewing, walk-through) and tools (e.g., text-editors, software debuggers, source code comparitors, language processors) need to be identified and the software design criteria be determined (HE92).

Quality assurance for code development and maintenance implies a systematic approach, starting with the careful formulation of code design objectives (section 3.2), criteria, and standards, followed by an implementation strategy. The implementation strategy includes the design of the code structure and a description of the way in which software engineering principles will be applied to the code. In this planning stage, measures are to be taken to ensure complete documentation of code design and implementation, record-keeping of the coding process, description of the purpose and structure of each code segment (functions, subroutines), and record-keeping of the code verification process.

Records for the coding and verification process may include a description of the fundamental algorithms describing the physical process(es) to be modeled; the means by which the mathematical algorithms have been translated into computer code (e.g., FORTRAN); results of discrete checks on the subroutines for accuracy; and comparisons among the codes' numerical solutions with either analytical or other independently verified numerical solutions.

Software used for compliance assessment should have both internal and external documentation. Internal documentation, which is part of the source code, describes the operation of the program and includes the name of the author, other sources of the software, and its revision history. External documentation includes a software abstract, an on-line help file stored on the applicable computer system, records of verification and changes, and formal reports including a theory manual and a user's manual.

Code verification or testing ensures that the underlying mathematical algorithms have been correctly translated into computer code. The verification process varies for different codes and ranges from simply checking the results of a plotting routine to comparing the results of the computer code to known analytical solutions or to results from other verified codes.

Traceability describes the ability of the performance assessment analyst to identify the software that was used to perform a particular calculation, including its name, date, and version number, while retrievability refers to the availability of the same version of the software for further use.

3.3.3.1 Code Documentation

In general, the code documentation should describe the theoretical framework represented by the model on which the code is based, code structure and language standards applied, and code use instructions regarding model setup and code execution parameters. The documentation should also include a complete treatment of the equations on which the model is based, the underlying mathematical and conceptual assumptions, the boundary conditions that are incorporated in the model, the method and algorithms used to solve the equations, and the limiting conditions resulting from the chosen approach. The documentation should also include user's instructions for implementing and operating the code and preparing data files. It should present examples of model formulation (e.g., grid design, assignment of boundary conditions), complete with input and output file descriptions, and include an extensive code verification and validation or field testing report.

Code Documentation Issues. An integral part of the code development process is the preparation of the code documentation. This documentation of QA in model development consists of reports and files pertaining to the development of the model and could include:

- A report on the development of the code including the (standardized and approved) programmer's bound notebook containing detailed descriptions of the code verification process;
- Verification report including verification scenarios, parameter values, boundary and initial conditions, source-term conditions, dominant flow and transport processes;
- Orientation and spacing of the grid and justification;
- Time-stepping scheme and justification;
- Changes and documentation of changes made in code after baselining;
- Executable and source code version of baselined code;
- Input and output (numerical and graphical) for each verification run;
- Notebook containing reference material (e.g., published papers, laboratory results, programmer's rationale) used to formulate the verification problem.

Furthermore, the purpose of the software documentation is to (GAS79):

- record technical information that enables system and program changes to be made quickly and effectively;

- enable programmers and system analysts, other than software originators, to use and to work on the programs;
- assist the user in understanding what the program is about and what it can do;
- increase program sharing potential;
- facilitate auditing and verification of program operations;
- provide managers with information to review at significant developmental milestones so that they may independently determine that project requirements have been met and that resources should continue to be expended;
- reduce disruptive effects of personnel turnover;
- facilitate understanding among managers, developers, programmers, operators, and users by providing information about maintenance, training, and changes in and operation of the software;
- inform other potential users of the functions and capabilities of the software, so that they can determine whether it serves their needs.

The user's information could consist of items such as:

- an extended model description;
- model input data description and format;
- type of output data provided;
- code execution preparation instructions;
- sample model runs;
- trouble shooting guide; and
- contact person/affiliated office.

The programmer's information could consist of items such as:

- model specifications;
- model description;
- flow charts;
- descriptions of routines;
- database description;
- source listing;
- error messages; and
- contact person/affiliated office.

The analyst's information could consist of items such as:

- a functional description of the model;
- model input and output data;
- code verification and validation information; and
- contact person/affiliated office.

The code itself should be well structured and internally well documented; where possible, self-explanatory parameter, variable, subroutine, and function names should be used.

3.3.3.2 Code Testing

Before a code can be used as a planning and decision-making tool, its credentials must be established through systematic testing of the model's correctness and evaluation of the model's performance characteristics (HEI89). Of the two major approaches available, the evaluation or review process is qualitative in nature, while code-testing results can be expressed using quantitative performance measures. Code testing (or code verification) is aimed at detecting programming errors, testing embedded algorithms, and evaluating the operational characteristics of the code through its execution on carefully selected example test problems and test data sets. ASME standard in NQA-2a-1990 Addenda (Part 2.7) defines software verification as the process of determining whether or not the product of a given phase of the software development cycle fulfills the requirements imposed by the previous phase.

It is necessary to distinguish between generic simulation codes based on an analytical solution of the governing equation(s) and codes that include a numerical solution. Verification of a coded analytical solution is restricted to comparison with independently calculated results using the same mathematical expression, i.e., manual calculations, using the results from computer programs coded independently by third-party programmers. Verification of a code formulated with numerical methods might take two forms: (1) comparison with analytical solutions, and (2) code intercomparison between numerically based codes, representing the same generic simulation model, using synthetic data sets.

It is also important to distinguish between code testing and model testing. Code testing is limited to establishing the correctness of the computer code with respect to the criteria and requirements for which it is designed (e.g., to represent the mathematical model). Model testing (or model validation) is more inclusive than code testing, as it represents the final step in determining the validity of the quantitative relationships derived for the real-world system the model is designed to simulate.

Attempts to validate models must address the issue of spatial and temporal variability when comparing model predictions with limited field observations. If sufficient field data are obtained to derive the probability distribution of contaminant concentrations, the results of a stochastic model can be compared directly. For a deterministic model, however, the traditional approach has been to vary the input data within its expected range of variability (or uncertainty) and determine whether the model results satisfactorily match historical field measured values. This code-testing exercise is sometimes referred to as history matching.

See Chapter 2 for additional Quality Assurance discussions.

3.3.4 Hardware Requirements

In general, hardware requirements are rarely a discriminatory factor in the selection of a computer code. However, a number of the codes that DOE intends to use in modeling the WIPP will require very sophisticated hardware, not so much because of the intrinsic requirements of the code but because the processes to be modeled are very complex.

3.3.5 Mathematical Solution Methodology

Every groundwater or contaminant transport model is based upon a set of mathematical equations. Solution methodology refers to the strategy and techniques used to solve these equations. In groundwater modeling, the equations are normally solved for head and/or contaminant concentrations. Other disposal system processes will be modeled with codes that solve for gas-filled porosities and the quantity of radioactive material (in curies) brought to the surface as cuttings generated by a drilling operation that penetrates the disposal system.

Analytical solutions are used in modeling investigations to solve many different kinds of problems. For example, aquifer parameters are obtained from aquifer pumping and tracer tests through the use of analytical models, and groundwater flow and contaminant transport rates can also be estimated with the use of analytical models.

Numerical models provide solutions to the differential equations describing room collapse, water movement, and solute transport using numerical methods such as finite differences and finite elements. Numerical methods account for complex geometry and heterogeneous media, as well as dispersion, diffusion, matrix deformation, salt creep and chemical retardation processes (e.g., sorption, precipitation, radioactive decay, ion exchange, degradation). These methods almost always require a digital computer, greater quantities of data than analytical modeling, and experienced modelers.

The validity of the results from mathematical models depends strongly on the quality and quantity of the input data. Stochastic, numerical, and analytical codes have strengths and weaknesses inherent within their formulations.

3.3.6 Code Dimensionality

The determination as to the number of dimensions that a code should be able to simulate is based primarily upon the modeling objectives and the dimensionality of the processes the code is designed to simulate.

In determining how many dimensions are necessary to meet the objectives, a basic understanding is needed of how the physical processes (e.g., salt creep, groundwater flow, transport, dose rate) are affected by the exclusion or inclusion of an additional dimension.

The movement of groundwater and contaminants is usually controlled by advective and dispersive processes which are inherently three-dimensional. Advection is more responsible for the time (i.e., travel time) it takes for a contaminant to travel from the source term to a downgradient receptor, while dispersion directly influences the concentration of the contaminant along its travel path. This fact is very important in that it provides an intuitive sense for the effect dimensionality has on contaminant migration rates and concentrations.

As a general rule, the fewer the dimensions, the more the model results will over-estimate concentrations and under-estimate travel times. In a model with fewer dimensions, predicted concentrations will generally be greater because dispersion, which is a three-dimensional process, will be dimension limited and will not occur to the same degree as it actually would in the field. Similarly, predicted travel times will be shorter than the actual travel time, not because of a change in the contaminant velocities but because a more direct travel path is assumed. Therefore, the lower dimensionality models tend to be more conservative in their predictions and are frequently used for screening analyses.

One-dimensional simulations of contaminant transport usually ignore dispersion altogether, and contamination is assumed to migrate solely by advection, which results in a highly conservative approximation. Vertical analyses in one dimension are generally reserved for evaluating flow and transport in the unsaturated zone. In the 1992 PA modeling of the Culebra Dolomite, advective and dispersive flow and transport were modeled in two dimensions with SECO, whereas matrix diffusion was confined to one dimension. This type of mixed-dimensionality approach is not uncommon early in a modeling analysis.

Two-dimensional analyses of an aquifer flow system can be performed as either a planar representation, where flow and transport are assumed to be horizontal (i.e., longitudinal and transverse components), or as a cross-section where flow and transport components are confined to vertical and horizontal components. In most instances, two-dimensional analyses are performed in an areal orientation, with the exception of the unsaturated zone, and are based on the assumption that most contaminants enter the saturated system from above and that little vertical dispersion occurs. However, a number of limitations accompany two-dimensional planar simulations. These include the inability to simulate multiple layers (e.g., aquifers and aquitards) as well as any partial penetration effects. Furthermore, because vertical components of flow are ignored, a potentially artificial lower boundary on contaminant migration has been automatically assumed which may or may not be the case.

A two-dimensional formulation of the flow system is frequently sufficient for the purposes of risk assessment provided that flow and transport in the contaminated aquifer are essentially horizontal. The added complexities of a site-wide, three-dimensional flow and transport simulation are often believed to outweigh the expected improvement in the evaluation of risk. Complexities include limited site-wide hydraulic head and lithologic data with depth and significantly increased computational demands.

Quasi three-dimensional analyses remove some of the limitations inherent in two-dimensional analyses. Most notably, quasi three-dimensional simulations allow for the incorporation of multiple layers; however, flow and transport in the aquifers are still restrained to longitudinal and transverse horizontal components, whereas flow and transport in the aquitards are even further restricted to vertical flow components only. Although partial penetration effects still cannot be accommodated in quasi three-dimensional analyses, this method can sometimes provide a good compromise between the relatively simplistic two-dimensional analysis and the complex, fully three-dimensional analysis. This is the case particularly if vertical movement of contaminants or recharge from the shallow aquifer through a confining unit and into a deeper aquifer is suspected.

Fully three-dimensional modeling generally allows both the geology and all of the dominant flow and transport processes to be described in three dimensions. This approach is usually the most reliable means of predicting groundwater flow and contaminant transport characteristics, provided that sufficient representative data are available for the site.

3.4 MODEL APPLICATION

The application of a generic simulation model to a site-specific problem is often called "model application" or "(computer/simulation) code application." The application of a generic model to site-specific conditions should follow a well-structured model application protocol. Such protocols are described by Mercer and Faust (MER81), van der Heijde et. al (HEI88), and Anderson and Woessner (AND92), among others. Quality assurance in these types of studies follows the same pattern as discussed for generic model development projects and consists of using appropriate data, data analysis procedures, modeling methodology and technology, administrative procedures and auditing. To a large extent, the quality of a modeling study is determined by the expertise of the modeling and quality assessment teams. The following discussion is consistent with procedures found in an EPA-sponsored publication (HEI92).

Quality assurance in code application addresses all facets of the modeling process, including such issues as:

- Historical review of code verification and benchmarking process;
- Correct and clear formulation of problems to be solved;
- Project description and objectives;
- Type of modeling approach to the project;
- Decision whether modeling is the best available approach and if so, that the selected code is appropriate and cost-effective;
- Conceptualization of system and processes, including hydrogeologic framework, boundary conditions, stresses, and controls;
- Detailed description of assumptions and simplifications, both explicit and implicit (to be subject to critical peer review);
- Data acquisition and interpretation;
- Code selection considerations, or justification for modifying an existing code or developing a new one;
- Model preparation (parameter selection, data entry or reformatting, gridding);
- Validity of the parameter values used in the model application;
- Protocols for parameter estimation and model calibration to provide guidance, especially for sensitive parameters;
- Level of information in computer output (variables and parameters displayed; formats; layout);
- Identification of calibration goals and evaluation of how well they have been met (e.g., root-mean square errors, etc.);
- Role of sensitivity analysis in evaluating parameter uncertainties and creating probability distributions;

- Post-simulation analysis (including verification of reasonableness of results, uncertainty analysis, and the use of manual or automatic data processing techniques, as for contouring);

- Establishment of appropriate performance targets which should characterize the limits of the data;

- Presentation and documentation of results;

- Evaluation of how closely the modeling results answer the questions raised by management.

QA for model application should include complete record-keeping of each step of the modeling process. The paper trail for QA should consist of reports and files addressing the following items:

- Assumptions and limitations;

- Parameter or input values and sources including rationale for their selection, range, and distribution;

- Boundary and initial conditions;

- Nature of grid and grid design justification;

- Changes and verification of changes made in code;

- Actual input used;

- Output of model runs and interpretation;

- Validation (or at least calibration) of model.

As is the case with code development QA, all data files, source codes, and executable versions of computer software used in the modeling study should be retained for auditing or post-project re-use (in hard-copy and, at higher levels, in digital form) including:

- Version of the source and executable image of the code used;

- Calibration input and output;

- Verification input and output;

- Application input and output (e.g., for each of the scenarios studied).

If the code used in the modeling study is modified, then the code should be tested again according to a standard testing protocol; the code should be subject to the full QA procedure for code development, including accurate record-keeping and reporting. All new input and output files should be saved for inspection and possible re-use together with existing files, records, codes, and data sets.

3.5 REFERENCES

AND92 Anderson, M.P., and W.W. Woessner, 1992. Applied Groundwater Modeling: Simulation of Flow and Advective Transport. Academic Press, Inc., San Diego, California.

ASM90 American Society of Mechanical Engineers (ASME), 1990. Quality Assurance Program Requirements for Nuclear Facilities. ASME NQA-2a-1990, PART 2.7, Quality Assurance Requirements of Computer Software for Nuclear Facility Applications, Am. Soc. of Mechanical Engineers, New York.

AST84 American Society for Testing and Materials (ASTM), 1984. Standard Practices for Evaluating Environmental Fate Models of Chemicals. Annual Book of ASTM Standards, E 978-84, Am. Soc. for Testing and Materials, Philadelphia, Pennsylvania.

BRY87 Bryant, J.L., and N.P. Wilburn, 1987. Handbook of Software Quality Assurance Techniques Applicable to the Nuclear Industry. NUREG/CR-4640, Off. of Nuclear Reactor Regulation, U.S. Nuclear Regulatory Commission, Washington, DC.

GAS79 Gass, S.I., 1979. Computer Model Documentation: A Review and an Approach. NBS Special Publ. 500-39, Inst. for Computer Science and Technology, Nat. Bur. of Standards, U.S. Dept. of Commerce, Washington, DC.

HEI88 van der Heijde, P.K.M., and M.S. Beljin, 1988. Model Assessment for Delineating Wellhead Protection Areas. EPA 440/6-88-002, Office of Groundwater Protection, U.S. Environmental Protection Agency, Washington, DC.

HEI89 van der Heijde, P.K.M., 1989. Quality Assurance and Quality Control in Groundwater Modeling. GWMI 89-04. Internat. Ground Water Modeling Center, Holcomb

Research Inst., Indianapolis, Indiana.

HEI92 van der Heijde, P.K.M., and O.A. Elnawawy, 1992. Compilation of Groundwater Models. GWMI 91-06. International Ground Water Modeling Center, Colorado School of Mines, Golden, Colorado.

MER81 Mercer, J.W. and C.R. Faust, 1981. Ground Water Modeling, Nat. Water Well Assoc., Dublin, Ohio.

TAY85 Taylor, J.K., 1985. What is Quality Assurance? In: J.K. Taylor and T.W. Stanley (eds.), Quality Assurance for Environmental Measurements, pp. 5-11. ASTM Special Technical Publication 867, Am. Soc. for Testing and Materials, Philadelphia, Pennsylvania.